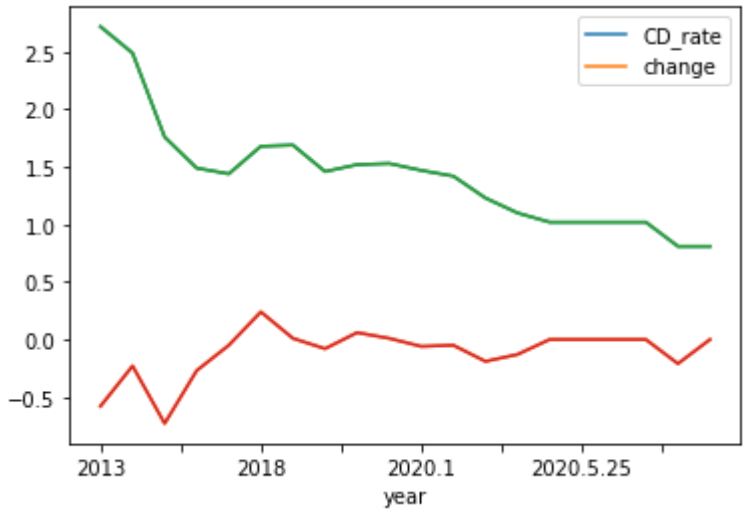
9. Matplotlib를 이용한 시각화

데이터를 그래프로 표현하기 1

```
import pandas as pd

df = pd.read_csv('bok_statistics_CD.csv', header=None)
print(df.head())
print('\n')

df.columns = ['year', 'CD_rate', 'change']
df.set_index('year', inplace=True)
print(df.head())
df.to_csv('bok_statistics_CD_2.csv')
print('\n')

df.plot()

df['CD_rate'].plot()
df['change'].plot()
```

데이터를 그래프로 표현하기 1


```
      0     1     2
0  2013  2.72 -0.58
1  2014  2.49 -0.23
2  2015  1.76 -0.73
3  2016  1.49 -0.27
4  2017  1.44 -0.05


      CD_rate  change
year
2013     2.72   -0.58
2014     2.49   -0.23
2015     1.76   -0.73
2016     1.49   -0.27
2017     1.44   -0.05
```

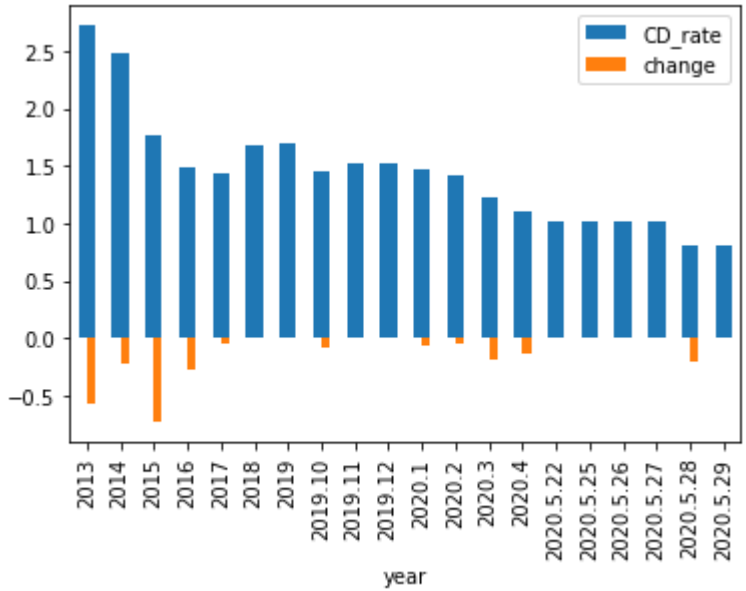


데이터를 그래프로 표현하기 2

```
import pandas as pd

df = pd.read_csv('bok_statistics_CD_2.csv', header=0, index_col=0)
print(df.head())
print('\n')

df.plot(kind='bar')

df['CD_rate'].plot(kind='bar')
#df['change'].plot(kind='bar')
```

데이터를 그래프로 표현하기 2

year	CD_rate	change
2013	2.72	-0.58
2014	2.49	-0.23
2015	1.76	-0.73
2016	1.49	-0.27
2017	1.44	-0.05

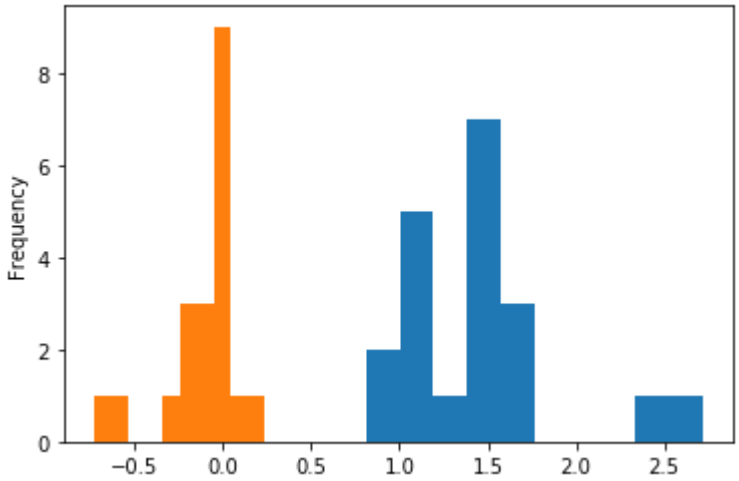


데이터를 그래프로 표현하기 3


```
import pandas as pd

df = pd.read_csv('bok_statistics_CD_2.csv', header=0, index_col=0)
print(df.head())
print('\n')

df['CD_rate'].plot(kind='hist')
df['change'].plot(kind='hist')
```

데이터를 그래프로 표현하기 3

year	CD_rate	change
2013	2.72	-0.58
2014	2.49	-0.23
2015	1.76	-0.73
2016	1.49	-0.27
2017	1.44	-0.05

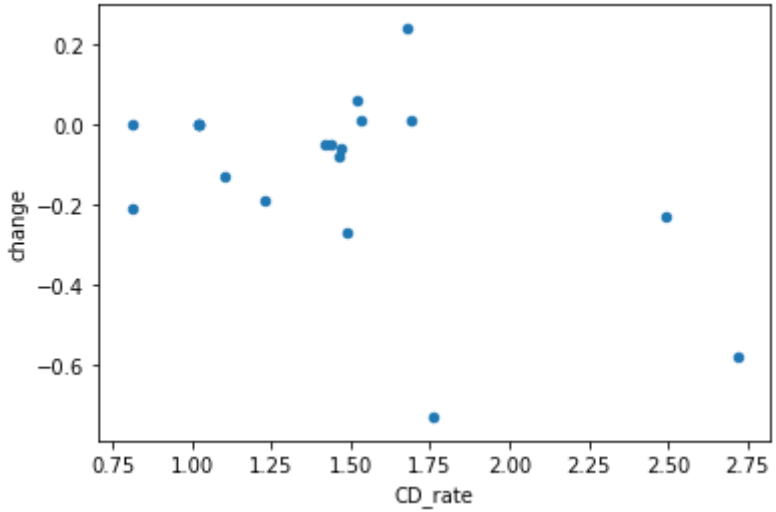


데이터를 그래프로 표현하기 4

```
import pandas as pd

df = pd.read_csv('bok_statistics_CD_2.csv', header=0, index_col=0)
print(df.head())
print('\n')

df.plot(x='CD_rate', y='change', kind='scatter')
```

데이터를 그래프로 표현하기 4

year	CD_rate	change
2013	2.72	-0.58
2014	2.49	-0.23
2015	1.76	-0.73
2016	1.49	-0.27
2017	1.44	-0.05

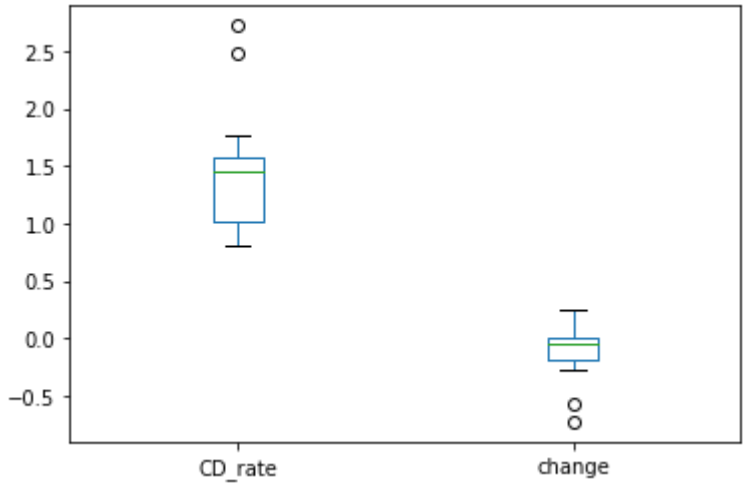


데이터를 그래프로 표현하기 5

```
import pandas as pd

df = pd.read_csv('bok_statistics_CD_2.csv', header=0, index_col=0)
print(df.head())
print('\n')

df.plot(kind='box')
```

데이터를 그래프로 표현하기 5

year	CD_rate	change
2013	2.72	-0.58
2014	2.49	-0.23
2015	1.76	-0.73
2016	1.49	-0.27
2017	1.44	-0.05

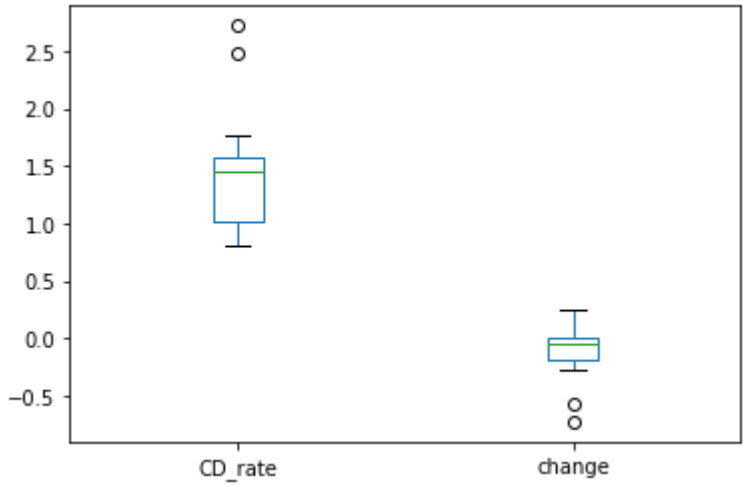


그래프를 그림파일로 저장하기


```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('bok_statistics_CD_2.csv', header=0, index_col=0)
print(df.head())
print('\n')

boxplot = df.plot(kind='box')
plt.savefig('boxplot.png')
```

그래프를 그림파일로 저장하기

year	CD_rate	change
2013	2.72	-0.58
2014	2.49	-0.23
2015	1.76	-0.73
2016	1.49	-0.27
2017	1.44	-0.05



9. Matplotlib를 이용한 시각화

9.1 Matplotlib I

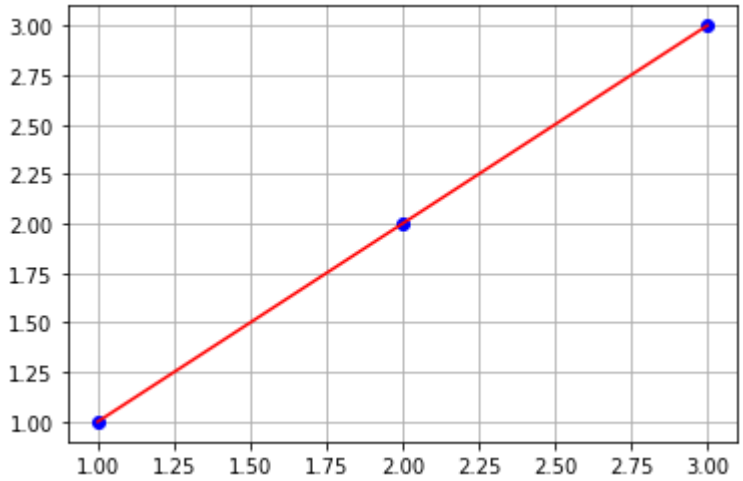
9.2 Matplotlib II

9.3 Matplotlib III

9.4 Matplotlib IV

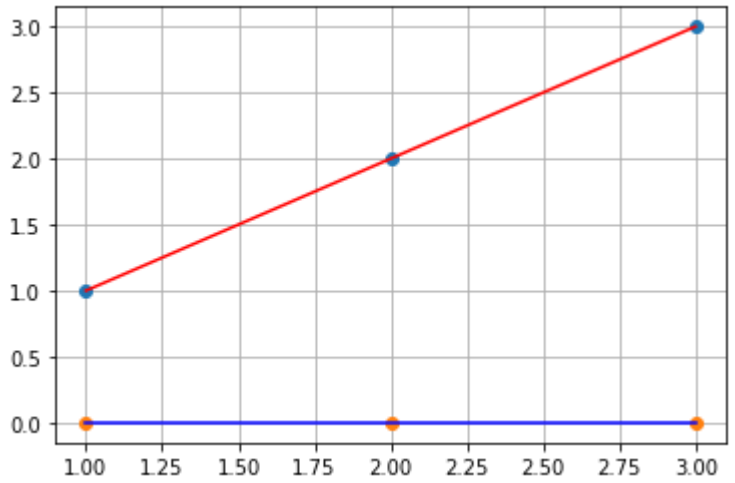
선형 그래프 그리기 1


```
import numpy as np
import matplotlib.pyplot as plt

x = np.array([1,2,3])
y = np.array([1,2,3])

plt.plot(x,y, 'bo' )
plt.plot(x,y, 'r-' )

plt.grid(True)
plt.show()
```



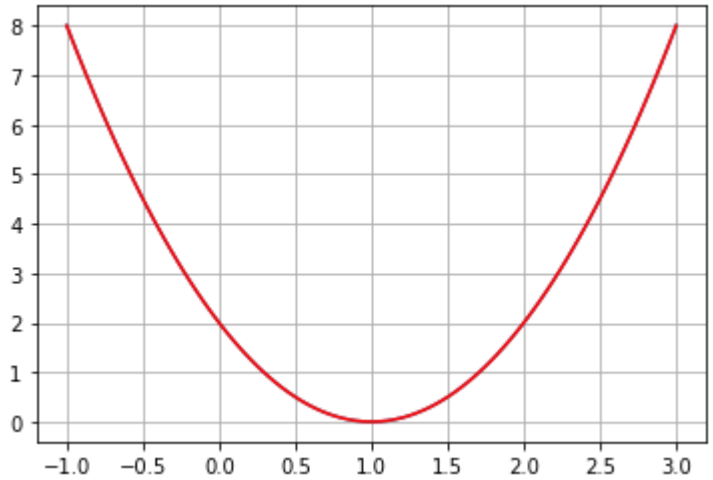
선형 그래프 그리기 2


```
import numpy as np
import matplotlib.pyplot as plt

x = np.array([1,2,3])
y = np.array([1,2,3])
w = 0
b = 0

y_hat = np.zeros(3)
for i in range(len(x)):
    y_hat[i] = w*x[i] + b;

plt.plot(x,y, 'o' )
plt.plot(x,y, 'r-' )

plt.plot(x,y_hat, 'o' )
plt.plot(x,y_hat, 'b-' )

plt.grid(True)
plt.show()
```



이차원 그래프 그리기


```
import numpy as np
import matplotlib.pyplot as plt

w = np.linspace(-1,3,100)
b = 0

j = np.zeros(100)

for i in range(len(w)):
    y_hat = w[i]*2 + b;
    j[i] = 0.5 * (y_hat - 2)**2

plt.plot(w,j, '-' )
plt.plot(w,j, 'r-' )

plt.grid(True)
plt.show()
```



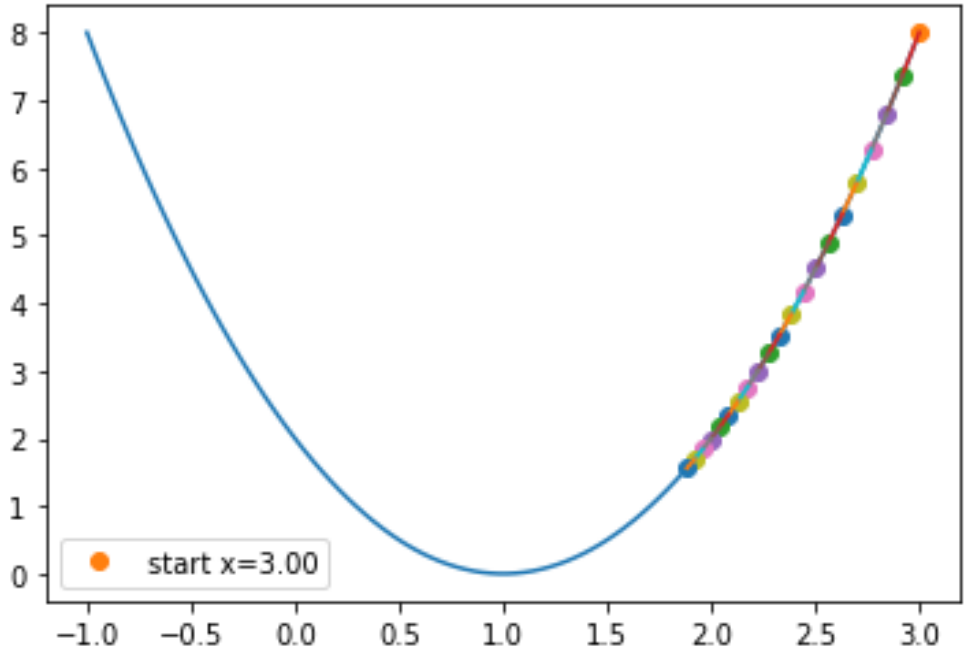
선형회귀 경사하강법 시각화


```
import numpy as np
import matplotlib.pyplot as plt

def f(x) :
    f = 0.5*(2-x*2)**2
    return f

def Df(x) :
    Df = 2*(2-x*2)
    return Df

def xp1(x,alpha) :
    xp1 = x + alpha * Df(x)
    return xp1
```

선형회귀 경사하강법 시각화


```
def plot_steps( guess, alpha, nsteps) :
    fig, ax = plt.subplots()
    x = np.linspace(-1,3,100)
    ax.plot(x, f(x))
    x = guess
    ax.plot(x,f(x), 'o', label='start x=%.2f' %x )
    for i in range(nsteps):
        xold = x
        x = xp1(x,alpha)
        ax.plot(x,f(x), 'o')
        ax.plot([xold,x],[f(xold),f(x)], '-')
    plt.legend()
    plt.show()

plot_steps( 3, 0.01, 20 )
```

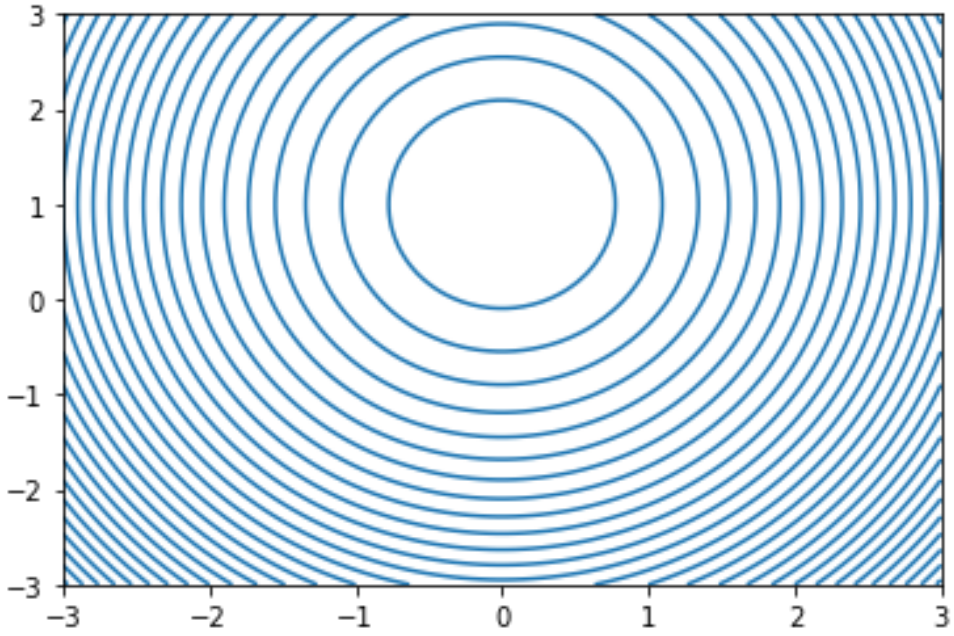
선형회귀 경사하강법 시각화



등고선 그래프 그리기


```
import matplotlib.pyplot as plt
import numpy as np

def J(a0, a1, x, y, m):
    ret=0
    for i in range(m):
        ret += 0.5*((a0 + a1*x[i]) - y[i] )**2
    return ret/m

x = np.linspace(-1,1,5)
y = x

a0 = np.linspace(-3,3,100)
a1 = np.linspace(-3,3,100)
aa0, aa1 = np.meshgrid(a0, a1)
plt.contour(aa0,aa1,J(aa0,aa1,x,y,m=len(x)) , colors='C0',
            levels=[i for i in np.arange(0,80,0.3)])
plt.show()
```

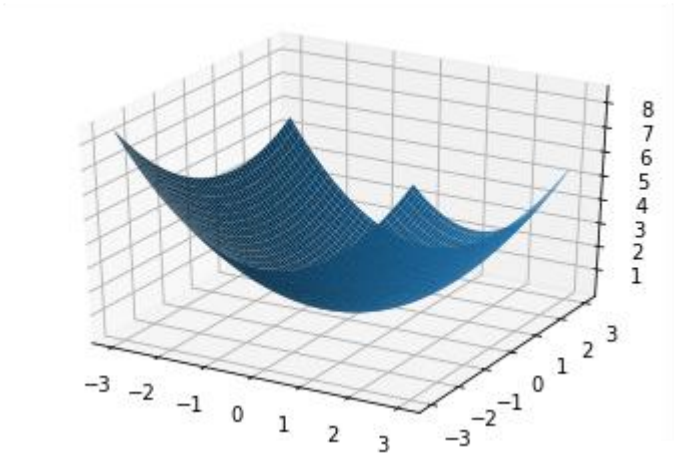
등고선 그래프 그리기



```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d.axes3d import Axes3D

fig = plt.figure()
ax = fig.add_subplot(1,1,1,projection='3d')

def pprint(arr):
    print("type:{}".format(type(arr)))
    print("shape: {}, dimension: {}, dtype:{}".format(arr.shape, arr.ndim,
arr.dtype))
    print("Array's Data:\n", arr)

def J(a0, a1, x, y, m):
    ret=0
    for i in range(m):
        ret += 0.5*((a0 + a1*x[i]) - y[i] )**2
    return ret/m
```

3 차원 그래프 그리기


```
x = np.linspace(-1,1,5)
y = x
a0 = np.linspace(-3,3,100)
a1 = np.linspace(-3,3,100)

aa0, aa1 = np.meshgrid(a0, a1)
ax.plot_surface(aa0, aa1, J(aa0,aa1,x,y,m=len(x)))

plt.show()
```



9. Matplotlib를 이용한 시각화

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt

text = open('usa_president_message.txt', encoding='UTF-8').read()

wordcloud = WordCloud(background_color='white',
                      width=1920,
                      height=1080).generate(text)

fig = plt.figure()
plt.imshow(wordcloud, interpolation='bilinear', cmap='YlOrBr')
plt.axis('off')

plt.savefig('usa_president_message_wordcloud.svg')
```

워드 클라우드 1


We
will
America
American
people
nation
one
many
first
protected
citizens
wealth
never
today
Together
new
job
great
will be
right
our
country
the world
Thank you
everyone
every
now
dream
President
will make
make america
day
all
across
heart
families
foreign
power
borders
bring back
God bless
factories
Politicians
millions

워드 클라우드 2


```
from konlpy.tag import Hannanum
from collections import Counter
from wordcloud import WordCloud
import matplotlib.pyplot as plt

text = open('2018_president_message.txt', encoding='cp949').read()

engin = Hannanum()
nouns = engin.nouns(text)
nouns = [n for n in nouns if len(n) > 1]

count = Counter(nouns)
tags = count.most_common(50)

wordcloud = WordCloud(font_path='c:/Windows/Fonts/malgun.ttf',
                      background_color='white',
                      width=1200,
                      height=800).generate_from_frequencies(dict(tags))
```

워드 클라우드 2

```
fig = plt.figure()
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')

plt.savefig('2018_president_message_wordcloud.svg')
```

워드 클라우드 3


```
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
from PIL import Image
import numpy as np

text = open('usa_president_message.txt', encoding='UTF-8').read()

image_path = 'usa_map.jpg'
usa_map = np.array(Image.open(image_path))

wordcloud = WordCloud(background_color='white',
                      max_font_size = 100,
                      max_words=1000,
                      stopwords=STOPWORDS,
                      mask=usa_map).generate(text)
```

워드 클라우드 3


```
fig = plt.figure(figsize=(15,15))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')

plt.savefig('usa_president_message_wordcloud_with_map.svg')
```

데이터프레임의 요약정보 확인하기 1


```
import pandas as pd

df = pd.read_csv('bok_statistics_CD.csv', header=None)

print(df.head())
print('\n')
print(df.info())
```

데이터프레임의 요약정보 확인하기 1

```
     0     1     2
0  2013  2.72 -0.58
1  2014  2.49 -0.23
2  2015  1.76 -0.73
3  2016  1.49 -0.27
4  2017  1.44 -0.05


<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   0       20 non-null     object
 1   1       20 non-null     float64
 2   2       20 non-null     float64
dtypes: float64(2), object(1)
memory usage: 608.0+ bytes
None
```

데이터프레임의 요약정보 확인하기 2


```
import pandas as pd

df = pd.read_csv('bok_statistics_CD.csv', header=None)

print(df.head())
print('\n')
print(df.describe())
```

데이터프레임의 요약정보 확인하기 2


```
 0     1     2
0  2013  2.72 -0.58
1  2014  2.49 -0.23
2  2015  1.76 -0.73
3  2016  1.49 -0.27
4  2017  1.44 -0.05


                1          2
count  20.000000  20.000000
mean    1.435000  -0.113000
std     0.494214   0.219236
min     0.810000  -0.730000
25%     1.020000  -0.195000
50%     1.450000  -0.050000
75%     1.567500   0.000000
max     2.720000   0.240000
```

9. Matplotlib를 이용한 시각화

레이더 차트

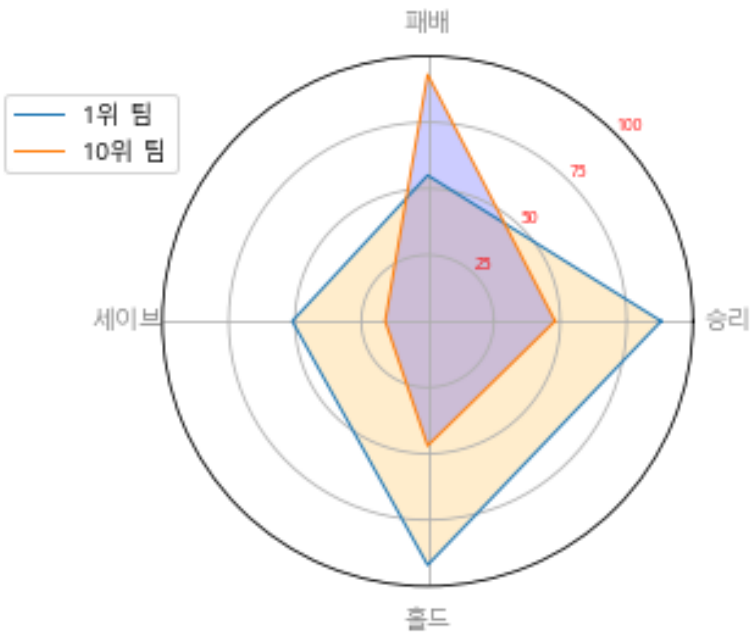

```
import matplotlib.pyplot as plt
import pandas as pd
from math import pi

kbo = pd.read_csv("kbo.csv")
print(kbo.head())
print("\n")

var = kbo.columns.to_list()[1:]
print(var)
print("\n")

val1 = kbo.loc[0, :].drop('팀명').values.tolist()
val1 += val1[:1]
print(val1)
print("\n")

val2 = kbo.loc[1, :].drop('팀명').values.tolist()
val2 += val2[:1]
print(val2)
print("\n")
```

```
num_var = len(var)
deg = [n / float(num_var) * 2 * pi for n in range(num_var)]
deg += deg[:1]
print(deg)
print("\n")

from matplotlib import font_manager, rc
font_path = "malgun.ttf"
font_name = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font_name)

ax = plt.subplot(111, polar=True)
plt.xticks(deg[:-1], var, color='grey', size=10)

ax.set_rlabel_position(45)
plt.yticks([25,50,75,100], ["25","50","75","100"], color="red", size=7)
plt.ylim(0,100)

ax.plot(deg, val1, linewidth=1, linestyle='solid', label='1위 팀')
ax.fill(deg, val1, 'orange', alpha=0.2)

ax.plot(deg, val2, linewidth=1, linestyle='solid', label='10위 팀')
ax.fill(deg, val2, 'blue', alpha=0.2)

plt.legend(loc='best', bbox_to_anchor=(0.05, 0.95))
```

레이더 차트


```
 팀명   승리  패배  세이브  홀드
0   1위 팀  88  55   51  92
1  10위 팀  48  93   16  47


['승리', '패배', '세이브', '홀드']


[88, 55, 51, 92, 88]


[48, 93, 16, 47, 48]


[0.0, 1.5707963267948966, 3.141592653589793, 4.71238898038469, 0.0]
```

레이더 차트



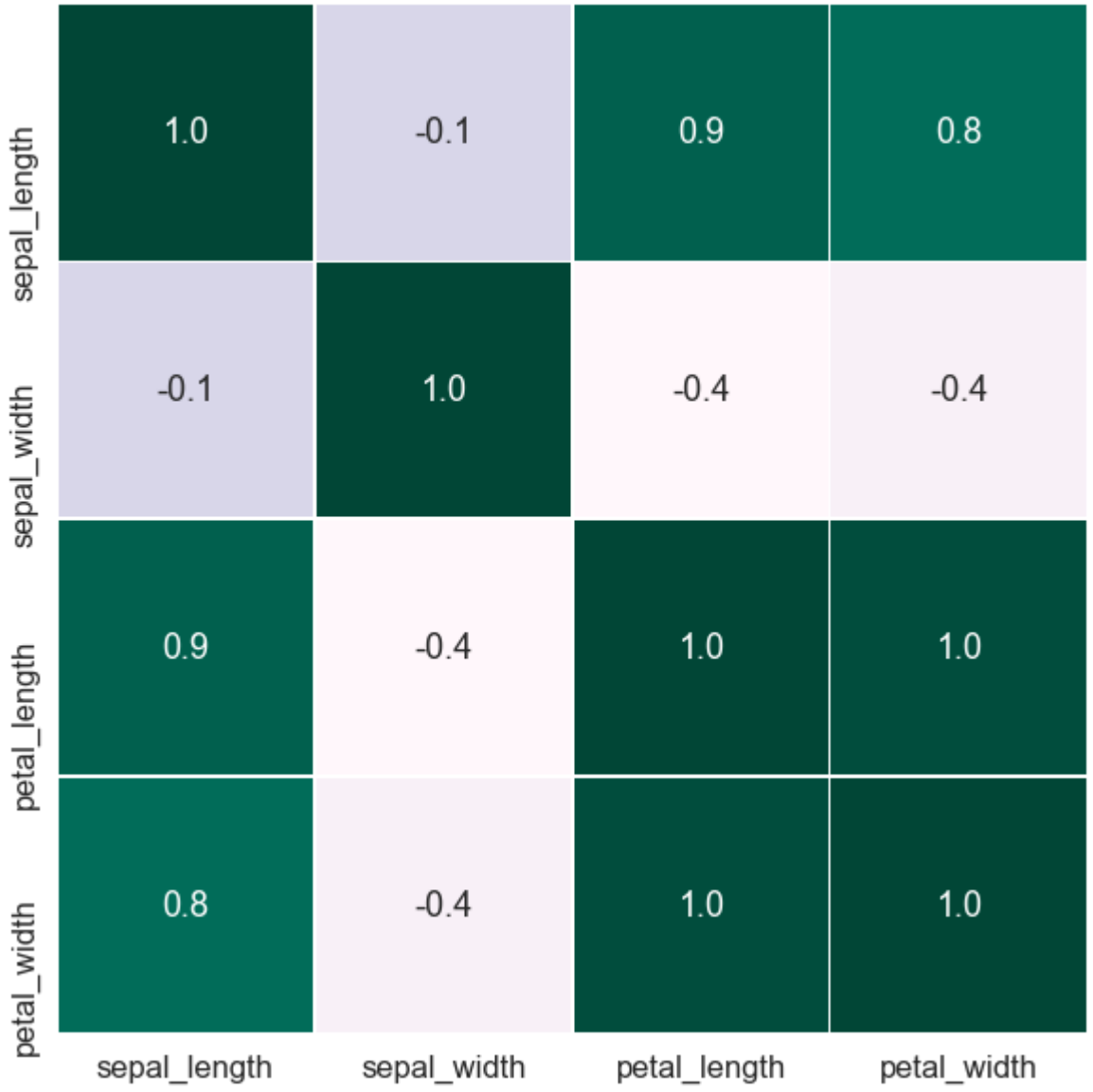
히트맵(heatmap)

```
import matplotlib.pyplot as plt
import seaborn as sns

df = sns.load_dataset('iris')
print(df.head())
print("\n")
print(df.columns.values)

plt.figure(figsize=(10,10))
corr = df.loc[:,'sepal_length':'petal_width'].corr()

sns.set(font_scale=1.5)
sns.heatmap(corr,
            annot=True,
            cmap='PuBuGn',
            fmt='.1f',
            square=True,
            linewidth=0.5,
            cbar=False)

plt.show()
```

히트맵(heatmap)

```
   sepal_length  sepal_width  petal_length  petal_width species
0           5.1          3.5           1.4          0.2  setosa
1           4.9          3.0           1.4          0.2  setosa
2           4.7          3.2           1.3          0.2  setosa
3           4.6          3.1           1.5          0.2  setosa
4           5.0          3.6           1.4          0.2  setosa


['sepal_length' 'sepal_width' 'petal_length' 'petal_width' 'species']
```

히트맵(heatmap)


sepal_length
sepal_width
petal_length
petal_width
1.0
-0.1
0.9
0.8
-0.1
1.0
-0.4
-0.4
0.9
-0.4
1.0
1.0
0.8
-0.4
1.0
1.0
sepal_length
sepal_width
petal_length
petal_width

채색 달력 그래프(calmap)


```
import pandas as pd
import calmap
import matplotlib.pyplot as plt

pd.set_option('display.max_columns', 20)

df = pd.read_excel('kospi.xls', parse_dates=['년/월/일'])
print(df.head())
print("\n")

df.columns = ['date', 'price', 'up_down', 'change', 'start', 'high', 'low',
              'vol_num', 'vol_amt', 'mkt_cap']
df = df.set_index('date', drop=True)
print(df.head())
print("\n")
```

채색 달력 그래프(calmap)


```
plt.figure(figsize=(16,8))
calmap.calendarplot(df.change,
                    monthticks=1, daylabels='MTWTFSS', dayticks=[0, 2, 4, 6],
                    cmap='YlGn', linewidth=0.05, fillcolor='grey',
                    fig_kws=dict(figsize=(14, 6)),
                    yearlabel_kws=dict(color='black', fontsize=12),
                    subplot_kws=dict(title='2018 KOSPI Price Trend'),
                    )

plt.show()
```

채색 달력 그래프(calmap)


```
 년/월/일        종가     대비  등락률(%)        시가        고가        저가  거래량(천주)  \
0 2018-12-28  2,041.04  12.60    0.62  2,036.70  2,046.97  2,035.41  352,678
1 2018-12-27  2,028.44   0.43    0.02  2,032.09  2,035.57  2,021.39  398,021
2 2018-12-26  2,028.01 -27.00   -1.31  2,028.81  2,037.83  2,014.28  321,499
3 2018-12-24  2,055.01  -6.48   -0.31  2,050.38  2,059.94  2,046.18  285,275
4 2018-12-21  2,061.49   1.37    0.07  2,052.70  2,061.51  2,049.76  311,389

             거래대금(원)              상장시가총액(원)
0  4,120,695,824,217  1,343,971,857,985,694
1  5,351,003,742,272  1,335,555,861,715,532
2  5,424,078,195,801  1,336,757,289,211,058
3  3,843,849,185,884  1,352,900,455,817,700
4  5,492,537,998,707  1,357,352,795,408,644


               price  up_down  change     start      high       low  vol_num  \
date
2018-12-28  2,041.04    12.60    0.62  2,036.70  2,046.97  2,035.41  352,678
2018-12-27  2,028.44     0.43    0.02  2,032.09  2,035.57  2,021.39  398,021
2018-12-26  2,028.01   -27.00   -1.31  2,028.81  2,037.83  2,014.28  321,499
2018-12-24  2,055.01    -6.48   -0.31  2,050.38  2,059.94  2,046.18  285,275
2018-12-21  2,061.49     1.37    0.07  2,052.70  2,061.51  2,049.76  311,389

                      vol_amt               mkt_cap
date
2018-12-28  4,120,695,824,217  1,343,971,857,985,694
2018-12-27  5,351,003,742,272  1,335,555,861,715,532
2018-12-26  5,424,078,195,801  1,336,757,289,211,058
2018-12-24  3,843,849,185,884  1,352,900,455,817,700
2018-12-21  5,492,537,998,707  1,357,352,795,408,644
```

채색 달력 그래프(calmap)